CalPERS

File Transport Services (Internal)

Title	File Transport Services (Internal)
Project	
Author	
Initiated On	
Filename	
Keywords	
Comments	
Last Saved By	
Last Saved On	

Revision History

Date	Version	Description	Author
01/01/08	0.1	Initial version	Middleware
08/10/08	2.0	Update revision	Shan Bains
10/21/08	2.1	Sample directory structure	Shan Bains

1. INTRODUCTION

The File Transport Service is designed to allow internal applications to move files from one FTP location (source) to another FTP location (target) using the FTP protocol. The service uses a standardized file naming convention and provides an audit trail and email notification of successful file transfers.

This document outlines the requirements for a CalPERS internal application to utilize this service.

Requirements

Each internal application will be provided with two folders on an internal FTP CalPERS server. The first folder will be a prod-out folder for transfer of files "outbound" TO another location. The second folder will be a prod-in folder for receipt of files "inbound" FROM another location.

The application logins will be restricted to only those folders within a specified Business Process. A Business Process is defined as a logical grouping of data and/or functionality as defined by Middleware.

1.1.1 Outbound File Processing:

- The application will upload files to the pre-determined prod-out folder on CalPERS internal ftp server using Binary mode.
- Data file names must be all lower case and must adhere to the file name convention described later in this document.
- Two files must be uploaded for each transaction, one data file and one semaphore file. Data files will have a .dat or .xml file extension. Sempahore files will have a .sem file extension. The semaphore file will have the same name as the data file but with a .sem file extension. The semaphore file is an empty file that indicates that the data file is complete and ready for further processing. Example of a file pair of files sent for each transaction. "*filename*.dat and "*filename*".sem or "*filename*.xml and "*filename*".sem
- The CalPERS File Transfer Service will retrieve the data file and the semaphore files from the FTP location prod-out directory at a pre-determined interval.
- The data file and the semaphore file will be deleted from the FTP location prod-out directory after successful processing. Non-conforming data files and semaphore files will not be processed from the source location.
- An Email message will be sent to a distribution list when the files are successfully processed and delivered to the target FTP location.

1.1.2 Inbound File Processing:

- Files coming from a sending application will be deposited in the pre-determined prod-in folder
- Data file names will be all lower case.
- Two files will be sent to the prod-in directory for each inbound transaction, one data file and one semaphore file. Data files will have a .dat or .xml file extension. Sempahore files have a .sem file extension. The semaphore file is an empty file that indicates that the data file is

complete and ready for further processing. Example of a file pair of files sent for each transaction.
"*filename*.dat and "*filename*".sem or "*filename*.xml and "*filename*".sem

- The application can poll the prod-in FTP location looking for files that are ready to be processed. Files are ready for processing when there is a pair of files with the same name but one has a .dat or .xml extension and one has a .sem extension. (see example above).
- The application will process the .dat or .xml file and then rename it to a .fin file extension. The renaming of the data file will indicate that the file has been processed and can be deleted by the File Transfer Cleanup Service.

1.2 File Naming Convention

Both inbound and outbound files must adhere to the file naming convention described below.

The standard format for file names:

A) ***yyyymmddhhmiss_sss_p(n).xxx***

Where:

yyyy is the year

mm is the month

dd is the day.

hh is the hours using a 24 hour clock

mi is the minutes

ss is the seconds

sss is the milliseconds, (use 000 if milliseconds can not be produced)

p(n) application specific area of the file name (project defined)

xxx is the file extension (.dat / .xml) for data files

1.3 Sample Directory Structures

The following examples are meant to illustrate the current Middleware FTP folder standards. It is recognized that variations may occur due to specific interface requirements. In these cases an agreement will be reached between the parties on the FTP folder structure used.

Example of plain file exchange interfaces with a single partner on either side of the business process:

Interface 00001
Username & Password combination number 1

- /apps/sftp/exsftpd/projects/filerelay/psr/00001/avs/test-out → AVS will drop off files here
- /apps/sftp/exsftpd/projects/filerelay/psr/00001/avs/test-in → AVS will pick up files here

- /apps/sftp/exsftpd/projects/filerelay/psr/00001/psr/test-out → PSR will drop off files here
- /apps/sftp/exsftpd/projects/filerelay/psr/00001/psr/test-in → PSR will pick up files here

Interface 00016
Username & Password combination number 2

- /apps/sftp/exsftpd/projects/filerelay/psr/00016/avs/test-out → AVS will drop off files here
- /apps/sftp/exsftpd/projects/filerelay/psr/00016/avs/test-in → AVS will pick up files here

- /apps/sftp/exsftpd/projects/filerelay/psr/00016/psr/test-out → PSR will drop off files here
- /apps/sftp/exsftpd/projects/filerelay/psr/00016/psr/test-in → PSR will pick up files here

For both of these examples, the directory at the partner level will be the login partner indicator for an interface with only one target partner. The test-in folder will only be applicable for the receiving partner based on the direction of the interface.

Example of a file exchange interface with multiple target partners on one side of the interface, e.g. ANSI 834 which deals with multiple health carriers:

Interface 50043
In the event that the interface has multiple target partners, the directory at the partner level will have the target directory partner name.

- /apps/sftp/exsftpd/projects/filerelay/psr/50043/blueshield/test-out → PSR will drop blueshield files here
- /apps/sftp/exsftpd/projects/filerelay/psr/50043/blueshield/test-in → This directory will not be used for the interface as currently defined

- /apps/sftp/exsftpd/projects/filerelay/psr/50043/bluecross/test-out → PSR will drop bluecross files here
- /apps/sftp/exsftpd/projects/filerelay/psr/50043/bluecross/test-in → This directory will not be used for the interface as currently defined

- /apps/sftp/exsftpd/projects/filerelay/psr/50043/kaiser/test-out → PSR will drop kaiser files here
- /apps/sftp/exsftpd/projects/filerelay/psr/50043/kaiser/test-in → This directory will not be used for the interface as currently defined

Example of a sub-project interface involving transformation, e.g. G2G Tlog:

Interface 00007
G2G Example directory structure (interface is bi-directional between SCO and PSR).

- /apps/sftp/exsftpd/projects/G2G/psr/00007/psr/test-out → PSR will drop files here
- /apps/sftp/exsftpd/projects/G2G psr/00007/psr/test- in → PSR will pick up files here

- /apps/sftp/exsftpd/projects/G2G/psr/00007/sco/test-out → SCO will drop files here
- /apps/sftp/exsftpd/projects/G2G/psr/00007/sco/test- in → SCO will pick up files here